Implementasi Enkripsi Dekripsi Algoritma Affine Cipher Berbasis Android

Encryption Implementation Of Deception Of Affine Cipher Algorithm Based On Android

Sasono Wibowo[1]
[1] Program Studi Sistem Informasi Universitas Dian Nuswantoro
[1] Jl. Nakula 1 No. 5 – 11, telp. (024) 3517261
e-mail : [1]sasono_wibowo@dsn.dinus.ac.id

Abstract

Dalam komunikasi antar orang pasti memiliki pembicaraan informasi yang bersifat privat atau orang lain tidak boleh tahu tentang pembicaraan yang terjadi. Keamanan sangat diperlukan untuk menjaga kerahasiaan informasi pada saat komunikasi. Masyarakat lebih sering menggunakan komunikasi dengan telepon seluler karena dinilai mudah dibawa dan tidak repot menggunakannya. Kriptografi yang biasa dikenal sebagai ilmu yang mempelajari bagaimana cara menyembunyikan pesan bisa diterapkan dalam aplikasi pada telepon seluler sebagai contoh smartphone android. Dengan mengimplementasikan algoritma affine cipher maka aplikasi yang akan dibuat bisa mengubah isi pesan yang ada dan dapat mengamankan informasi yang ada. Algoritma affine cipher merupakan perkembangan dari algoritma caesar dimana algoritma affine cipher menggunakan dua kunci. Dengan mengimplementasikan algoritma affine cipher ke dalam android maka diharapkan kita bisa menyimpan informasi dari siapapun tanpa terbaca.

Kata Kunci : Kriptografi, Affine Cipher, android, Implementasi, Informasi

Abstract

In communication between people must have a private information talk or someone else should not know about the conversation that happened. Security is necessary to maintain the confidentiality of information at the time of communication. Communities more often use communication with mobile phones because it is considered easy to carry and do not bother using it. Cryptography commonly known as the science of learning how to hide messages can be applied in applications on mobile phones as examples of android smartphones. By implementing affine cipher algorithm, the application will be made can change the contents of existing messages and can secure the existing information. The affine cipher algorithm is the development of a cesarean algorithm in which the affine cipher algorithm uses two keys. By implementing affine cipher algorithm into android then hopefully we can save information from anyone without read.

Keywords: Cryptography, Affine Cipher, android, Implementation, Information

1. PENDAHULUAN

Perkembangan teknologi komunikasi bisa dilihat dari alat komunikasi berupa mesin fax, mesin telegram, telepon, pager, telepon seluler, dll. Dengan adanya teknologi tersebut membuat orang tidak mengenal jarak dan waktu untuk terus berkomunikasi. Dalam berkomunikasi pasti ada halnya suatu informasi tersebut sangat penting dan rahasia. Komunikasi secara visual atau dengan teks bisa dibilang tingkat keamanannya masih kurang. Dilihat dari apakah pesan tersebut akan dibaca orang lain atau tidak. Untuk mengirimkan pesan yang bernilai penting dan rahasia, dibutuhkan keamanan dalam teks tersebut

Masalah pengiriman pesan ini biasanya terdapat pada suatu instansi baik negeri maupun swasta, misalnya saja ada seorang karyawan bagian rekrutmen disuatu perusahaan ingin mengirimkan pesan ke bagian HRD, karena pesan yang dikirim bersifat rahasia maka dibutuhkan pengenkripsian pesan tersebut.

Pengenkripsian data atau informasi sangatlah penting guna menunjang keamanan informasi dalam suatu instansi baik negeri maupun swasta, karena bisa memberikan jaminan keamanan pesan yang akan diberikan kepada orang atau lembaga yang dituju. Oleh sebab itu, enkripsi sangatlah dibutuhkan bagi user (pengguna) jika ingin data atau informasi yang dimilikinya terjamin kerahasiaannya.

2. METODE PENELITIAN

Kriptografi adalah suatu ilmu yang mempelajari bagaimana cara menjaga agar data atau pesan tetap aman saat dikirimkan, dari pengirim ke penerima tanpa mengalami gangguan dari pihak ketiga. Menurut Bruce Scheiner dalam bukunya "Applied Cryptography", kriptografi adalah ilmu pengetahuan dan seni menjaga pesan (informasi) agar tetap aman (secure).

Kriptografi itu sendiri terdiri dari dua proses utama yakni proses enkripsi dan proses dekripsi. Seperti yang telah dijelaskan di atas, proses enkripsi mengubah *plaintext* menjadi *ciphertext* (dengan menggunakan kunci tertentu) sehingga isi informasi pada pesan tersebut sukar dimengerti.

Peranan kunci sangatlah penting dalam proses enkripsi dan dekripsi (disamping pula algoritma yang digunakan) sehingga kerahasiaannya sangatlah penting, apabila kerahasiaannya terbongkar, maka isi dari pesan dapat diketahui.

Secara matematis, proses enkripsi merupakan pengoperasian fungsi E (enkripsi) menggunakan e (kunci enkripsi) pada P *(plaintext)* sehingga dihasilkan C *(ciphertext)*, notasinya

$$E_e(P) = C$$

Sedangkan untuk proses dekripsi, merupakan pengoperasian fungsi D (dekripsi) menggunakan d (kunci dekripsi) pada C *(ciphertext)* sehingga dihasilkan P *(plaintext)*, notasinya

$$D_d(C) = P$$

Sehingga dari dua hubungan diatas berlaku :

$$D_d(E_e(P)) = P$$

2.1 Affine Cipher

Affine cipher adalah perluasan dari metode *Caesar cipher* yang menggunakan teknik substitusi yang menggunakan fungsi liniear *ap+b* untuk enkripsi teks asli *p* dan $a^{-1}c$-*b* untuk dekripsi teks sandi *c* pada Z_{26}. Kunci pada sandi *Affine* adalah 2 integer yaitu *a* dan *b*. Nilai *a* yang dapat dipakai adalah anggota elemen pada Z_{26} yang memiliki invers yaitu yang memenuhi gcd(a,26) = 1.

2.1.1 Proses Enkripsi Affine Cipher

Affine cipher merupakan sandi yang bekerja secara subtitusi. Pada *affine cipher* terdapat abjad sejumlah *m*, yang yaitu rentang *m*-1, maksudnya adalah awal abjad yaitu huruf "A" bernilai 0, huruf kedua "B" bernilai 1, dan seterusnya hingga huruf terakhir dalam abjad yaitu huruf "Z" bernilai 25.

Adapun rumus enkripsi dengan menggunakan affine cipher pada satu huruf plaintext menjadi satu huruf ciphertext adalah sebagai berikut:

$$E(x) = (ax + b) \bmod m,$$

Dimana m adalah ukuran abjad, ini berarti modulus m adalah modulus dari ukuran abjad, sedangkan jumlah abjad dalam rentang affine cipher adalah 25, maka modulus m adalah modulus 25. Sedangkan a adalah bilangan yang harus dipilih secara bebas, namun memiliki syarat haruslah coprime dengan nilai m, artinya harus memiliki nilai faktor yang positif.

2.1.2 Proses Dekripsi Affine Cipher

Fungsi dekripsi affine cipher adalah:

$$D(x) = a\text{-}1(x\text{-}b) \bmod m,$$

a-1 adalah invers perkalian a modulus m. Yaitu, memenuhi persamaan:

$$1 = aa\text{-}1 \bmod m.$$

Invers perkalian a hanya ada jika a dan m adalah coprime. Jika tidak maka proses algoritma akan terhenti.

2.2 Eclipse

Eclipse adalah sebuah IDE (Integrated Development Environment) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform. Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya. Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

2.3 Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih

bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana. Aplikasi-aplikasi berbasis java umumnya dikompilasi ke dalam bytecode dan dapat dijalankan pada berbagai Java Virtual Machine (JVM).

2.4 Android

Android adalah sebuah sistem operasi untuk perangkat mobile berbasis linux yang mencakup sistem operasi, middleware dan aplikasi. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel/smartphone. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance.

3. HASIL PEMBAHASAN

3.1 Analisis Kebutuhan

Aplikasi Cryssage ini digunakan untuk mengirim dan menerima pesan. Cryssage akan mengenkripsi pesan yang akan dikirim menjadi *ciphertext* dan Cryssage akan mendekripsi pesan masuk berupa *ciphertext* menjadi *plaintext*.

3.2 Context Diagram

Gambar 1. Context Diagram Aplikasi Cryssage

Gambar 1 menjelaskan pengirim menginputkan pesan, password dan nomor tujuan kepada sistem. Sistem menghasilkan output berupa laporan pesan terkirim kepada pengirim. Penerima mendapatkan ciphertext dan nomor pengirim. Untuk dapat membaca pesan, penerima menginputkan password kepada sistem dan sistem memberikan output berupa pesan kepada penerima.

3.3 Data Flow Diagram

Pada gambar 2 sistem akan dipecah menjadi proses-proses kecil sehingga dapat menjelaskan proses-proses dan arus data yang mengalir dalam sistem. Proses-proses yang terdapat pada gambar 2 adalah:

1. Mengubah ke ASCII

   Proses ini mengubah pesan dan password ke dalamkodeASCII.

2. Enkripsi

Proses ini melakukan pengenkripsian pesan menggunakan algoritma affine cipher dengan kunci/password yang diinputkan.

3. Pengiriman pesan

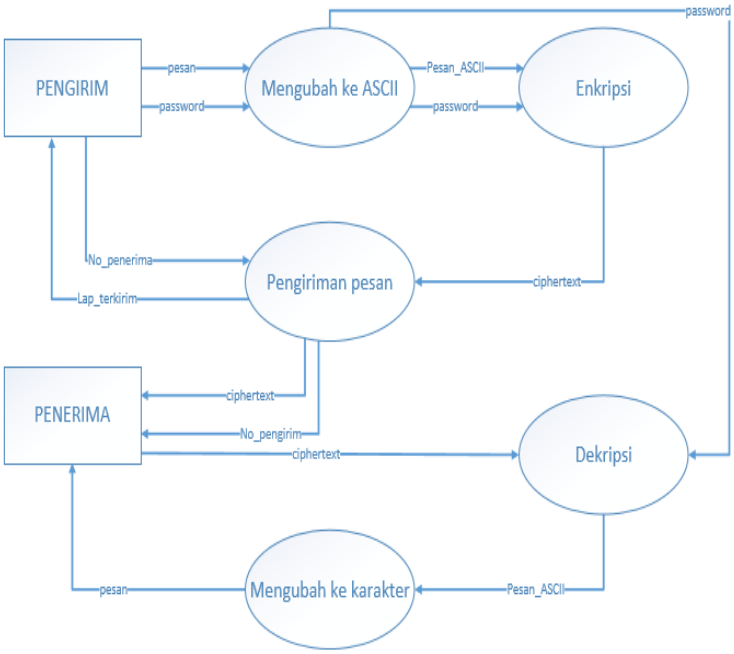
   Proses ini mengirimkan pesan yang telah dienkripsi dan memberikan laporan pengiriman kepada pengirim bahwa pesan telah terkirim ke nomor yang telahdiinputkan.

4. Dekripsi

   Proses ini melakukan pendekripsian pesan sesuai dengan password yang diinputkan. Apabila password benar maka ciphertext akan menjadi pesan asli. Jika password salah pesan akan tetap didekripsi akan tetapi pesan yang didapat bukan pesan asli. Karena key yang dipakai untuk mendekripsi ciphertext salah.

5. Mengubah ke karakter

   Proses ini mengubah kode ASCII yang diterima dari hasil dekripsi ke dalam karakter menggunakan password.



Gambar 2. Data Flow Diagram Aplikasi Cryssage

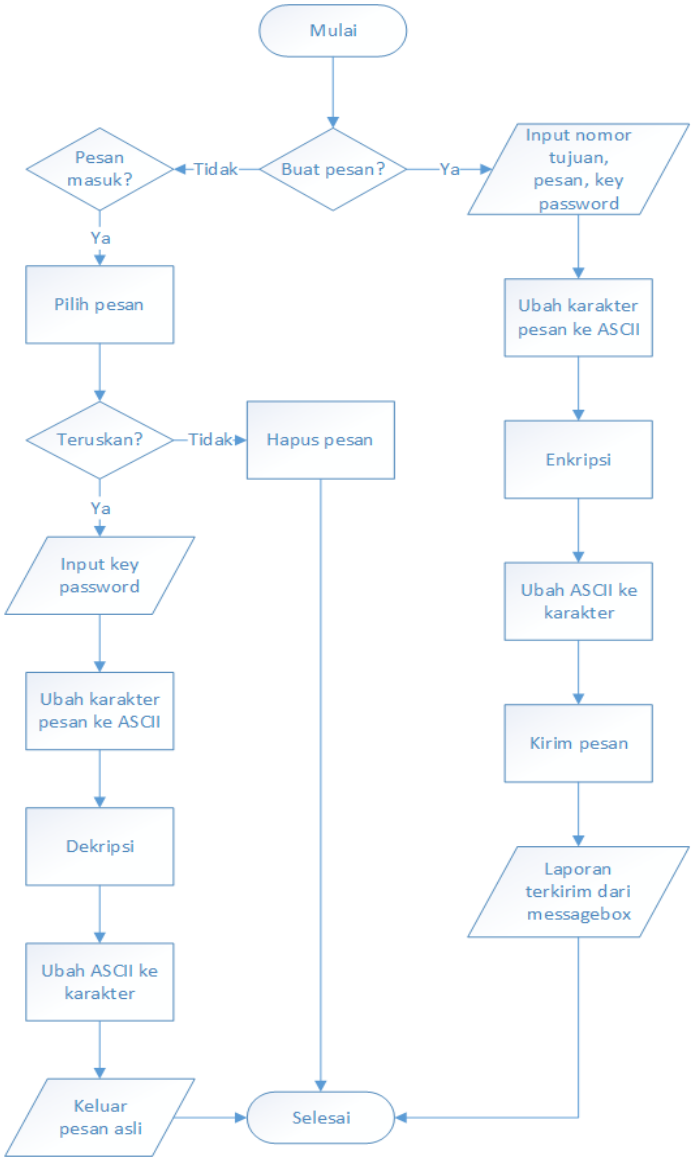
3.4 Perancangan Sistem

Perancangan sistem merupakan deskripsi proses-proses yang akan dilaksanakan dalam sebuah rancangan desain tampilan sebelum dimulai pembuatan *code* atau *coding*. Aplikasi Cryssage memiliki empat kelas yaitu: MainActivity, BuatPesan, DataPesan, dan LihatPesan. Proses *coding* dibuat menggunakan aplikasi eclipse. Fungsi masing-masing kelas sebagai berikut:

1. MainActivity, kelas ini merupakan kelas utama yang menghubungkan kelas yang lain dan kelas yang pertama ditemui saat menjalankan aplikasi Cryssage.

2. BuatPesan, kelas ini tempat terletaknya proses enkripsi dekripsi pesan dan tempat proses pengiriman pesan terjadi.
3. DataPesan, kelas ini menyimpan data dari pesan yang masuk dan pesan yang keluar.
4. LihatPesan, kelas ini menampilkan pesan yang masuk dan pesan yang keluar secara spesifik.

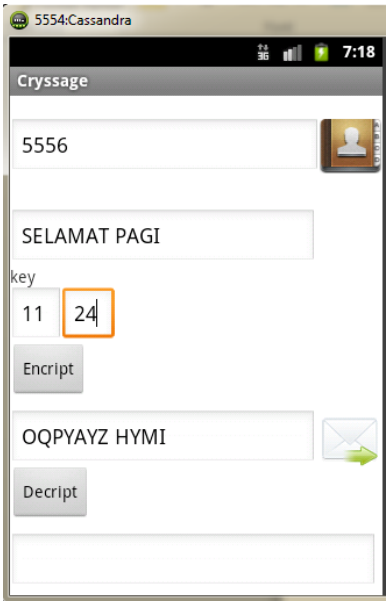
3.5 Perancangan Interface



Gambar 3. Flowchart Aplikasi Cryssage

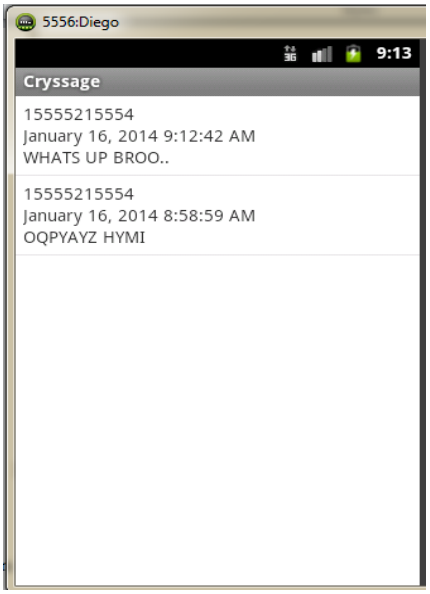
Perancangan *interface* adalah prosesperancangan form-form tampilan layar. Selain ituproses ini juga ditentukan bentuk dan isi dokumen sumber untuk memasukkan data yang kemudian akan diolah menjadi keluaran yang dapat digunakan oleh pengguna.

Langkah pertama password yang diinputkan merupakan key yang akan dipakai untuk mengenkripsi pesan.



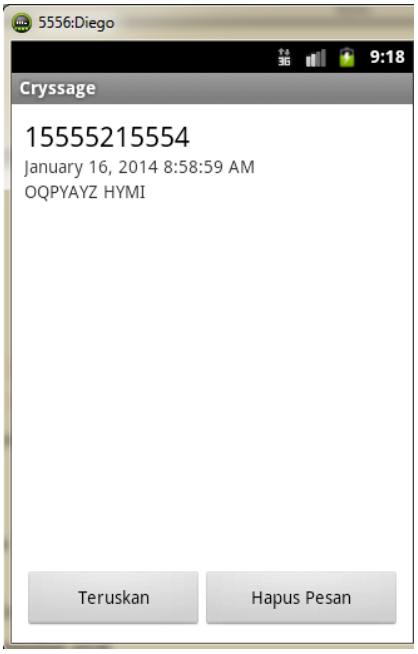
Gambar 4. Proses Pengiriman dan Enkripsi Pesan

Selanjutnya pesan yang masuk akan muncul di tampilan listpesan. Pada proses ini semua pesan yang diterima bisa dilihat di listpesan dan akan terhubung ke lihatpesan saat user memilih pesan yang ingin dibaca.



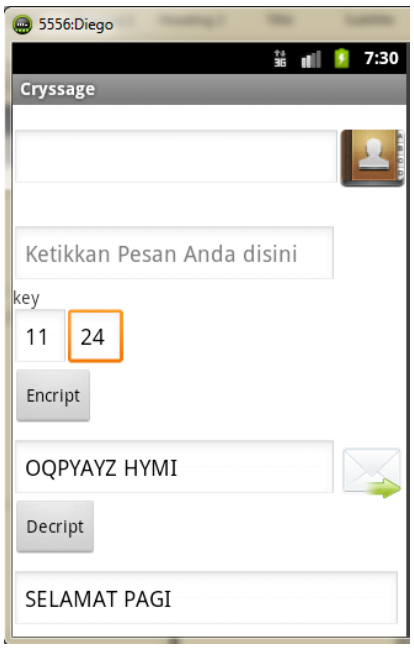
Gambar 5. Kotak Pesan Masuk Aplikasi Cryssage

Dari pesan yang masuk User dapat memilih pesan yang masuk dari beberapa pesan masuk yang ada sehingga user dapat membaca pesan secara detail. Pesan yang dipilih dapat diteruskan untuk melakukan proses dekripsi pesan atau pesan dapat dihapus.



Gambar 6. Pemilihan Pesan

Pesan yang dipilih user dapat dilakukan proses dekripsi pesan dengan memasukkan key password yang benar sehingga pesan asli akan muncul.



Gambar 7. Proses Dekripsi Pesan

3.6 Perhitungan Manual

- Proses Enkripsi

Pesan : SELAMATPAGI

ASCII : 83 69 76 65 77 65 84 80 65 71 73

Konversi rumus (ASCII – 65) menjadi nilai array

Nilai array : 18 4 11 0 12 0 19 15 0 6 8

Key 1 = 11

Key 2 = 24

P1 = 18 ➔C1 = 11.18 + 24 = 222 (mod 26) = 14

P2 = 4 ➔C2 = 11.4 + 24 = 68 (mod 26) = 16

P3 = 11 ➔C3 = 11.11 + 24 = 145 (mod 26) = 15

P4 = 0 ➔C4 = 11.0 + 24 = 24 (mod 26) = 24

P5 = 12 ➔C5 = 11.12 + 24 = 156 (mod 26) = 0

P6 = 0 ➔C6 = 11.0 + 24 = 24 (mod 26) = 24

P7 = 19 ➔C7 = 11.19 + 24 = 233 (mod 26) = 25

P8 = 15 ➔ C8 = 11.15 + 24 = 189 (mod 26) = 7

P9 = 0 ➔ C9 = 11.0 + 24 = 24 (mod 26) = 24

P10 = 6 ➔ C10 = 11.6 + 24 = 90 (mod 26) = 12

P11 = 8 ➔ C11 = 11.8 + 24 = 112 (mod 26) = 8

Nilai array : 14 16 15 24 0 24 25 7 24 12 8

Konversi rumus (nilai array + 65) menjadi ASCII

ASCII : 79 81 80 89 65 89 90 72 89 77 73

Ciphertext : OQPYAYZHYMI

- Proses dekripsi

Ciphertext : O Q P Y A Y Z H Y M I

ASCII : 79 81 80 89 65 89 90 72 89 77 73

Konversi rumus (ASCII – 65) menjadi nilai array

Nilai array : 14 16 15 24 0 24 25 7 24 12 8

Key 1 = 11

Key 2 = 24

C1 = 14 ➔ P1 = 19.((14 + 26)–24) = 304 (mod 26) = 18

C2 = 16 ➔ P2 = 19.((16 + 26) – 24) = 342 (mod 26) = 4

C3 = 15 ➔ P3 = 19.((15 + 26) – 24) = 323 (mod 26) = 11

C4 = 24 ➔ P4 = 19.((24 + 26) – 24) = 494 (mod 26) = 0

C5 = 0 ➔ P5 = 19.((0 + 26) – 24) = 38 (mod 26) = 12

C6 = 24 ➔ P6 = 19.((24 + 26) – 24) = 494 (mod 26) = 0

C7 = 25 ➔ P7 = 19.((25 + 26) – 24) = 513 (mod 26) = 19

C8 = 7 ➔ P8 = 19.((7 + 26) – 24) = 171 (mod 26) = 15

C9 = 24 ➔ P9 = 19.((24 + 26) – 24) = 494 (mod 26) = 0

C10 = 12 ➔ P10 = 19.((12 + 26) – 24) = 266 (mod 26) = 6

C11 = 8 ➔ P11 = 19.((8 + 26) – 24) = 190 (mod 26) = 8

Nilai array : 18 4 11 0 12 0 19 15 0 6 8

Konversi rumus (nilai array + 65) menjadi ASCII

ASCII : 83 69 76 65 77 65 84 80 65 71 73

Pesan : SELAMAT PAGI

4. KESIMPULAN

Algoritma yang dibuat mengunakan kombinasi kunci yang sulit terprediksi, dikarenakan menggunakan kombinasi dua kunci yang berbeda dan Aplikasi Cryssage ini bisa digunakan untuk melakukan enkripsi pesan dan mengirimnya ke nomor tujuan penerima pesan.

Aplikasi Cryssage bisa digunakan oleh user dalam lingkup umum yang membutuhkan keamanan informasi melalui sms dan mencegah orang yang tidak berkenan untuk mengetahui informasi yang telah dikirim user kepada penerima.

5. SARAN

Untuk perbaikan dan pengembangan aplikasi Cryssage lebih lanjut, disarankan aplikasi Cryssage didesain dengan tampilan yang lebih menarik atau Aplikasi dapat ditambahkan menu help atau bantuan untuk memudahkan user menggunakan

DAFTAR PUSTAKA

[1] December, John. 1997. *Presenting Java Inilah Java*. Prenhallindo.

[2] Forouzan, Behrouz A. 2009. *Cryptography and network security*.Mcgraw-hill.

[3] Hamdani. Kriptografi menggunakan metode affine. http://hamdani.blog.ugm.ac.id/2011/07/07/kriptografi-untuk-text-message-menggunakan-metode-affine. Tanggal akses 25 Desember 2017.

[4] Mkyong. How to convert character to ascii in java. http://www.mkyong.com/java/how-to-convert-character-to-ascii-in-java. Tanggal akses 17 Desember 2017.

[5] Rauf, ruzlan akba. Kode ascii lengkap. http://informatikakba-ruzlan.blogspot.com/2013/05/kode-ascii-lengkap.html. Tanggal akses 14 Desember 2017.

[6] Sadikin, Rifki. 2012. *Kriptografi Untuk Keamanan Jaringan*. Andi.

[7] Purwito, Heru. Contoh aplikasi sms sederhana pada android.

http://herupurwito.wordpress.com/2013/04/11/contoh-aplikasi-sms-sederhana-pada-android. Tanggal akses 7 November 2017.

[8] Sommerville, Ian, 2003. *Rekayasa Perangkat Lunak*. Erlangga. Jakarta.

[9] Munawar. 2012. *Perancangan Algoritma Sistem Keamanan Data Menggunakan* Metode *Kriptografi Asimetris*. Jurnal Komputer dan Informatika (KOMPUTA). Volume 1. Edisi I.

[10] Y. Kurniawan. 2004. *Kriptografi Keamanan Internet dan Jaringan Komunikasi*. Informatika. Bandung.

[11] B Schneier. 1996. *Applied Cryptography*. John Wiley and Sons. Inc.NY

[12] T. Heriyanto. 1999. *Pengenalan Kriptografi*. Internet.

[13] Kristanto. 2003. *Keamanan Data Pada Jaringan Komputer*. Gava Media. Yogayakarta.